SYSTEM AND METHOD OF FACE RECOGNITION USING PROPORTIONS OF LEARNED MODEL

BACKGROUND OF THE INVENTION

Field of the Invention

The present invention relates to face recognition systems and particularly, to a system and method for performing face recognition using proportions of the learned model.

Discussion of the Prior Art

Existing face recognition systems attempt to recognize an unknown face by matching against prior instances of that subject's face(s). This is typically performed by training a classifier against prior instances of a subject's face and then using the trained classifier to identify the subject by matching against new instances of that subjects face. As known, training a classifier involves learning a model of the subject's face. Existing systems use the whole model during classification.

While the ultimate goal in the design of any pattern recognition system is to achieve the best possible classification (predictive) performance, this objective traditionally has led to the development of different classification schemes for any pattern recognition problem to be solved. The results of an experimental assessment of the different designs would then be the basis for choosing

one of the classifiers (model selection) as a final solution to the problem. It has been observed in such design studies, that although one of the designs would yield the best performance, the sets of patterns misclassified by the different classifiers would not necessarily overlap as recognized by Kittler J., Hatef, H. and Duin, R. P. W. in the reference entitled "Combining Classifiers, in Proceedings of the 13th International Conference on pattern Recognition", Vol. II, pp. 897-901, Vienna, Austria, 1996. This suggested that different classifier designs potentially offered complementary information about the patterns to be classified, which could be harnessed to improve the overall performance.

It had been a common practice in the application of neural networks to train many different candidate networks and then select the best, on the basis of performance on an independent validation set for instance, and to keep only this network and to discard the rest. There are two disadvantages with such an approach. First, all of the effort involved in training the remaining networks is wasted. Second, the generalization performance on the validation set has a random component due to the noise in the data, and so the network which had best performance on the validation set might not be the one with the best performance on new or unseen test data. These drawbacks can be overcome by combining the networks together to form a committee of networks. The importance of such an approach is that it can lead to significant improvements in the predictions on new data, while involving little additional computational effort. In fact

the performance of a committee can be better than the performance of the best single network used in isolation as recognized by Bishop C. M., in the reference entitled "Neural Networks for Pattern Recognition," Oxford Press, Oxford, UK, pp. 364-377, 1997.

In order to recognize faces, recognition systems have employed multiple classifiers each trained on profiles of an individual face. On presentation of a probe (test image), the probe is matched with each of the learned model and the scores obtained from each classifier are used up to arrive at a consensus decision. An obvious disadvantage of training multiple classifiers is that a lot of time and space is wasted in training and storing the model files.

It would be highly desirable to provide a face recognition system and methodology whereby instead of having multiple classifiers trained on various profiles of an individual face, a single classifier may be trained on either a frontal face or multiple profiles of an individual's face.

It would further be highly desirable to provide a face recognition system and method wherein proportions of a subject's model is implemented and used to match against different proportions of a subject's face. That is, during testing, an unknown facial image is identified by matching different proportions of the learned model and the unknown facial image.

SUMMARY OF THE INVENTION

Accordingly, it is an object of the present invention to provide a system and method implementing a

classifier (e.g., RBF networks) that may be trained to recognize either a frontal face or multiple profiles of an individual's face.

It is a further object of the present invention to provide a face recognition system and method implementing a single classifier device that has been trained on a subject's frontal profile of the face and, during testing, taking an unknown test image and match it against the learned model using different proportions.

Preferably, after matching against each proportion, a probability of match is determined and the scores are then combined to arrive at a consensus decision. For example, each proportion classified will generate a vote. That is, if ten (10) proportions are used, 10 votes would be obtained. Then, a simple voting rule (e.g., if six (6) out of ten (10) are for 'A' then the identity of the subject is 'A') is used to ascertain the identity of the individual.

In accordance with the principles of the invention, there is provided a system and method for classifying facial image data, the method comprising the steps of: training a classifier device for recognizing one or more facial images and obtaining corresponding learned models the facial images used for training; inputting a vector including data representing a portion of an unknown facial image to be recognized into the classifier; classifying the portion of the unknown facial image according to a classification method; repeating inputting and classifying steps using a different portion of the unknown facial image at each iteration; and, identifying a

single class result from the different portions input to the classifier.

Advantageously, although an RBF classifier may be used, it is understood that one could use other methods as well, including combinations of various probabilistic/stochastic methods.

BRIEF DESCRIPTION OF THE DRAWINGS

Details of the invention disclosed herein shall be described below, with the aid of the figures listed below, in which:

Figure 1 generally illustrates the architecture of a traditional three-layer back-propagation network 10 according to which an RBF network implemented in accordance with the principles of the present invention is structured;

Figure 2 illustrates a sample set of facial images fed to the network.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

For purposes of description, a Radial Basis Function ("RBF") classifier is implemented although any classification method/device may be implemented. A description of an RBF classifier device is available from commonly-owned, co-pending Unites States Patent Application Serial No. 09/794,443 entitled CLASSIFICATION OF OBJECTS THROUGH MODEL ENSEMBLES filed February 27, 2001, the whole contents and disclosure of which is incorporated by reference as if fully set forth herein.

The construction of an RBF network as disclosed in commonly-owned, co-pending Unites States Patent Application Serial No. 09/794,443, is now described with reference to Figure 1. As shown in Figure 1, the basic RBF network classifier 10 is structured in accordance with a traditional three-layer back-propagation network 10 including a first input layer 12 made up of source nodes (e.g., k sensory units); a second or hidden layer 14 comprising i nodes whose function is to cluster the data and reduce its dimensionality; and, a third or output layer 18 comprising j nodes whose function is to supply the responses 20 of the network 10 to the activation patterns applied to the input layer 12. The transformation from the input space to the hidden-unit space is *non-linear*, whereas the transformation from the hidden-unit space to the output space is *linear*. In particular, as discussed in the reference to C. M. Bishop, Neural Networks for Pattern Recognition, Clarendon Press, Oxford, 1997, the contents and disclosure of which is incorporated herein by reference, an RBF classifier network 10 may be viewed in two ways: 1) to interpret the RBF classifier as a set of kernel functions that expand input vectors into a high-dimensional space in order to take advantage of the mathematical fact that a classification problem cast into a high-dimensional space is more likely to be linearly separable than one in a low-dimensional space; and, 2) to interpret the RBF classifier as a function-mapping interpolation method that tries to construct hypersurfaces, one for each class, by taking a linear combination of the Basis Functions (BF). These hypersurfaces may be viewed as

discriminant functions, where the surface has a high value for the class it represents and a low value for all others. An unknown input vector is classified as belonging to the class associated with the hypersurface with the largest output at that point. In this case, the BFs do not serve as a basis for a high-dimensional space, but as components in a finite expansion of the desired hypersurface where the component coefficients, (the weights) have to be trained.

In further view of Figure 1, the RBF classifier 10, connections 22 between the input layer 12 and hidden layer 14 have unit weights and, as a result, do not have to be trained. Nodes in the hidden layer 14, i.e., called Basis Function (BF) nodes, have a Gaussian pulse nonlinearity specified by a particular mean vector μ_i (i.e., center parameter) and variance vector σ_i^2 (i.e., width parameter), where i = 1, … , F and F is the number of BF nodes. Note that σ_i^2 represents the diagonal entries of the covariance matrix of Gaussian pulse (i). Given a D-dimensional input vector **X**, each BF node (i) outputs a scalar value y_i reflecting the activation of the BF caused by that input as represented by equation 1) as follows:

$$y_i = \phi_i\left(\|X - \mu_i\|\right) = \exp\left[-\sum_{k=1}^{D} \frac{\left(x_k - \mu_{ik}\right)^2}{2h\sigma^2_{ik}}\right], \qquad (1)$$

Where h is a proportionality constant for the variance, x_k is the k^{th} component of the input vector **X** = [x_1, x_2, … , x_D], and μ_{ik} and σ_{ik}^2 are the k^{th} components of the mean and

variance vectors, respectively, of basis node (i). Inputs that are close to the center of the Gaussian BF result in higher activations, while those that are far away result in lower activations. Since each output node 18 of the RBF network forms a linear combination of the BF node activations, the portion of the network connecting the second (hidden) and output layers is linear, as represented by equation 2) as follows:

$$z_j = \sum_i w_{ij} y_i + w_{oj} \qquad (2)$$

where z_j is the output of the j^{th} output node, y_i is the activation of the i^{th} BF node, w_{ij} is the weight 24 connecting the i^{th} BF node to the j^{th} output node, and w_{oj} is the bias or threshold of the j^{th} output node. This bias comes from the weights associated with a BF node that has a constant unit output regardless of the input.

An unknown vector **X** is classified as belonging to the class associated with the output node j with the largest output z_j. The weights w_{ij} in the linear network are not solved using iterative minimization methods such as gradient descent. They are determined quickly and exactly using a matrix pseudoinverse technique such as described in above-mentioned reference to C. M. Bishop, Neural Networks for Pattern Recognition, Clarendon Press, Oxford, 1997.

A detailed algorithmic description of the preferable RBF classifier that may be implemented in the present invention is provided herein in Tables 1 and 2. As shown in Table 1, initially, the size of the RBF network 10

is determined by selecting F, the number of BFs nodes. The appropriate value of F is problem-specific and usually depends on the dimensionality of the problem and the complexity of the decision regions to be formed. In general, F can be determined empirically by trying a variety of Fs, or it can set to some constant number, usually larger than the input dimension of the problem. After F is set, the mean μ_I and variance σ_I^2 vectors of the BFs may be determined using a variety of methods. They can be trained along with the output weights using a back-propagation gradient descent technique, but this usually requires a long training time and may lead to suboptimal local minima. Alternatively, the means and variances may be determined before training the output weights. Training of the networks would then involve only determining the weights.

The BF means (centers) and variances (widths) are normally chosen so as to cover the space of interest. Different techniques may be used as known in the art: for example, one technique implements a grid of equally spaced BFs that sample the input space; another technique implements a clustering algorithm such as **k**-means to determine the set of BF centers; other techniques implement chosen random vectors from the training set as BF centers, making sure that each class is represented.

Once the BF centers or means are determined, the BF variances or widths σ_I^2 may be set. They can be fixed to some global value or set to reflect the density of the data vectors in the vicinity of the BF center. In addition, a global proportionality factor *H* for the variances is

included to allow for rescaling of the BF widths. By searching the space of H for values that result in good performance, its proper value is determined.

After the BF parameters are set, the next step is to train the output weights w_{ij} in the linear network. Individual training patterns $X(p)$ and their class labels $C(p)$ are presented to the classifier, and the resulting BF node outputs $y_I(p)$, are computed. These and desired outputs $d_j(p)$ are then used to determine the $F \times F$ correlation matrix "**R**" and the $F \times M$ output matrix "**B**". Note that each training pattern produces one **R** and **B** matrices. The final **R** and **B** matrices are the result of the sum of N individual **R** and **B** matrices, where N is the total number of training patterns. Once all N patterns have been presented to the classifier, the output weights w_{ij} are determined. The final correlation matrix **R** is inverted and is used to determine each w_{ij}.

1. Initialize

(a) Fix the network structure by selecting *F*, the number of basis functions, where each basis function *I* has the output where *k* is the component index.

$$y_i = \phi_i\left(\left\|X - \mu_i\right\|\right) = \exp\left[-\sum_{k=1}^{D} \frac{\left(x_k - \mu_{ik}\right)^2}{2h\sigma^2_{ik}}\right],$$

(b) Determine the basis function means μ_I, where $I = 1, \ldots, F$, using K-means clustering algorithm.

(c) Determine the basis function variances σ_I^2, where $I = 1, \ldots, F$.

(d) Determine *H*, a global proportionality factor for the basis function variances by empirical search

2. Present Training

(a) Input training patterns *X(p)* and their class labels *C(p)* to the classifier, where the pattern index is $p = 1, \ldots, N$.

(b) Compute the output of the basis function nodes $y_I(p)$, where $I = 1, \ldots, F$, resulting from pattern *X(p)*.

$$R_{il} = \sum_p y_i(p) y_l(p)$$

(c) Compute the *F x F* correlation matrix R of the basis function outputs:

(d) Compute the *F x M* output matrix B, where d_j is the desired output and *M* is the number of output classes:

$$B_{lj} = \sum_p y_l(p) d_j(p), \text{ where } d_j(p) = \begin{cases} 1 \text{ if } C(p) = j \\ 0 \text{ otherwise} \end{cases},$$

and $j = 1, \ldots, M$.

3. Determine Weights

(a) Invert the *F x F* correlation matrix **R** to get $\mathbf{R}^{-1}$.

(b) Solve for the weights in the network using the following equation:

$$w^*_{ij} = \sum_l \left(R^{-1}\right)_{il} B_{lj}$$

Table 1

As shown in Table 2, classification is performed by presenting an unknown input vector $\mathbf{X}_{test}$ to the trained classifier and computing the resulting BF node outputs y_i. These values are then used, along with the weights w_{ij}, to compute the output values z_j. The input vector $\mathbf{X}_{test}$ is then classified as belonging to the class associated with the output node j with the largest z_j output.

1. Present input pattern portion $\mathbf{X}_{test}$ to the classifier
2. Classify a portion of $\mathbf{X}_{test}$
   (a) Compute the basis function outputs, for all F

$$y_i = \phi\left(\left\| X_{test} - \mu_i \right\|\right)$$

   basis functions

   (b) Compute output node activations:

$$z_j = \sum_i w_{ij} y_i + w_{oj}$$

   (c) Select the output z_j with the largest value and classify $\mathbf{X}_{test}$ portion as the class j;

   (d) Repeat steps 2(a)-2(c) using different proportions of decreased size.

Table 2

In the method of the present invention, the RBF input consists of **n**-size normalized facial gray-scale images fed to the network as one-dimensional, i.e., 1-D, vectors. The hidden (unsupervised) layer 14, implements an "enhanced" **k**-means clustering procedure, such as described in S. Gutta, J. Huang, P. Jonathon and H. Wechsler, Mixture of Experts for Classification of Gender, Ethnic Origin, and

Pose of Human Faces, IEEE Transactions on Neural Networks, 11(4):948-960, July 2000, the contents and disclosure of which is incorporated by reference as if fully set forth herein, where both the number of Gaussian cluster nodes and their variances are dynamically set. The number of clusters may vary, in steps of 5, for instance, from 1/5 of the number of training images to **n**, the total number of training images. The width σ_I^2 of the Gaussian for each cluster, is set to the **maximum** (the distance between the center of the cluster and the farthest away member - within class diameter, the distance between the center of the cluster and closest pattern from all other clusters) multiplied by an overlap factor **o**, here equal to 2. The width is further dynamically refined using different proportionality constants **h**. The hidden layer 14 yields the equivalent of a functional shape base, where each cluster node encodes some common characteristics across the shape space. The output (supervised) layer maps face encodings ('expansions') along such a space to their corresponding ID classes and finds the corresponding expansion ('weight') coefficients using pseudoinverse techniques. Note that the number of clusters is frozen for that configuration (number of clusters and specific proportionality constant **h**) which yields 100 % accuracy on ID classification when tested on the same training images.

According to the invention, the input vectors to be used for training are full facial images, for example the facial images 30 shown in Figure 2, each comprising a size of 64x72 pixels, for example. According to the invention, a single classifier (RBF network 10, is trained

with these full images. However, during actual testing, different proportions of the test image are tested against different proportions of the model. For instance, step 2 of the classification algorithm depicted in Table 2, is an iterative process that performs a subtraction of the unknown test image with a different portion of the learned model in each iteration. Training is on a full face a full image and an X_{test} (full image) may be input at the first iteration. A first output score is obtained, which includes a confidence (probability) measure, e.g., as illustrated as step 2(c) in Table 2, having a value between 0 and 1, and a label identifying the class label (learned model). At each iteration, these steps are repeated each time using a different percentage of the image, i.e., portions of the learned model. For example, in a next iteration, a smaller portion of the unknown image, e.g., 90%, may be compared against the corresponding 90% of the learned model image for each class, and so on. As a result of each comparison, a further a confidence (probability) measure and a label identifying the class (learned model) is determined by the classifier device. Thus, as indicated in Table 2, the whole of step 2(a) is in a loop with the process repeated any number of times depending upon the number of proportions desired. For example, as selectable by a user, the X_{test} image portions utilized may range from maximum (e.g., 100% of the full image) to minimum (e.g., 50% of the full image) at a 10% or 5% portion reduction at each iteration. As described in commonly-owned, co-pending U.S. Patent Application No. _____ [Attorney Docket 702052, D#14900] entitled SYSTEM AND METHOD OF FACE

RECOGNITION THROUGH 1/2 FACES, the whole disclosure and contents of which is incorporated by reference as if fully set forth herein, when the minimum image is used, i.e., 50%, it is imperative that at least one eye, 1/2 the nose and 1/2 the mouth of the facial image be captured, e.g., a vertical proportion of the image. The granularity of the portion reduction at each iteration may be a user selectable option and may depend on how good that data is and the computation cost consideration. It should be understood that a trade-off exists between the performance and cost. For instance, depending upon the level of security desired, i.e., the more secure the application, the finer granularity of proportion reduction at each iteration, and the greater number of comparisons will be performed at greater cost. For the case of 100% to 50% in with 10% image reduction proportions at each step there will be a total of six (6) confidence scores and class labels, whereby with 5% image reduction proportions at each step there will be a total of twelve (12) for each class. After the scores have been accumulated, rules may be applied to determine the class for that test image. For example, the scores may be combined to arrive at a consensus decision. One simple class may be majority rule however, more sophisticated rules may be applied, e.g., such as described in the reference to J. Kittler, M. Hateg, and R. P. W. Duin entitled "Combining Classifiers," Proc. of the 13th International Conference on Pattern Recognition, II: 897-901, Vienna, Austria, August 1996, the contents and disclosure of which is incorporated by reference herein. For example, each proportion classified

will generate a vote and if ten (10) proportions are used, 10 votes would be obtained. Then, a majority decision voting rule simple voting rule (e.g., if six (6) out of ten (10) are for 'A' then the identity of the subject is 'A') is used to ascertain the identity of the individual (class). In response, multiple votes are generated and, in the classifier, as shown in Figure 1, a selection device is 28 is provided with logic for applying voting rules to arrive at an appropriate decision.

While there has been shown and described what is considered to be preferred embodiments of the invention, it will, of course, be understood that various modifications and changes in form or detail could readily be made without departing from the spirit of the invention. It is therefore intended that the invention be not limited to the exact forms described and illustrated, but should be constructed to cover all modifications that may fall within the scope of the appended claims.